Specification

Title of the Invention

Internet Camera

Background of the Invention

The present invention relates to an Internet camera which is connected to the Internet and transmits image data files to a predetermined site of a file server on the Internet.

Conventionally, a digital camera has been widely used. The digital camera captures an image, converts the image to digital data, and store the digital data in a form of data files in a storage device such as a hard disk.

Recently, as an application of the digital camera, an Internet camera has been used. An example of the Internet camera is disclosed in United State Patent Application No. 09/204,289, the teachings of which are incorporated herein by reference.

The conventional Internet camera typically includes a network interface such as a modem for sending the data files via the Internet, and a data transfer client such as an FTP (File Transfer Protocol according to RFC 959) client

for controlling the network interface in order to transfer the data files stored in the storage device to a predetermined site of a file server such as an FTP server on the Internet.

In such an Internet camera, images are captured and the image data files are transmitted automatically, in accordance with a predetermined schedule.

Conventionally, however, the schedule includes only one script which defines a start time, an end time and an interval, and the camera operates in a similar manner everyday. The image capturing and/or transferring operations are repeatedly executed at the defined intervals during a period from the start time to the end time.

Since the schedule include only one script, the conventional Internet camera cannot capture the images according to a relatively complicated schedule such as a schedule extending in two days, e.g., a schedule for capturing images from 23:00 of a day to 1:00 of the next day.

Summary of the Invention

It is therefore an object of present invention for providing an Internet camera for capturing and transferring images according to a relatively complicated schedule as

described above.

For the above object, according to the invention, there is provided an Internet camera that captures images of an object and transmitting image data to a predetermined site on the Internet, which camera is provided with an image capturing device that captures images and converts the captured images into digital data, a storage device that stores digital data corresponding to the captured images in a form of image data files, respectively, a memory that stores schedule data which includes a plurality of schedule scripts, a network interface connected to the Internet, a data transfer client that controls the network interface to transfer the image data file to a predetermined site on the Internet, a schedule merging device that merges the plurality of schedule scripts into a merged schedule, and a controller that controls the image capturing device to capture images in accordance with the merged schedule, and controls the data transfer client to transfer the image data file corresponding to the captured image in accordance with the merged schedule to the predetermined site.

With the above configuration, since the image capturing operations and data transferring operations can be performed in accordance with a merged schedule which is generated by combining a plurality of schedule scripts,

even if each having a relatively simple data structure, a complicated scheduling can be achieved easily.

Optionally, the schedule merging device may assign priorities to the plurality of schedule scripts, respectively, procedures defined by a schedule script having a higher priority being executed if two or more schedule scripts overlap.

Further optionally, each of the plurality of schedule scripts may include a start time and an end time, between which the images are to be captured and the image data files are to be transferred, and a designation of days of a week, on which the images are to be captured and the image data files are to be transferred.

In this case, each of the plurality of schedule scripts may further include an interval, at every occurrence of which an image is to be captured and an image data file is to be transferred.

According to another aspect of the invention, there is provided an Internet camera that captures images of an object and transmitting image data to a predetermined site on the Internet, which camera is provided with an image capturing device that captures images and converts the captured images into digital data, a storage device that stores digital data corresponding to the captured images in a form of image data files, respectively, a memory that

stores schedule data which includes a plurality of schedule scripts, a schedule merging device that merges the plurality of schedule scripts into a merged schedule, and a controller that controls the image capturing device to capture images in accordance with the merged schedule.

With the above configuration, since the image capturing operations can be performed in accordance with a merged schedule, which is generated by combining a plurality of schedule scripts, a complicated scheduling can be achieved easily.

Optionally, the schedule merging device may assign priorities to the plurality of schedule scripts, respectively, procedures defined by a schedule script having a higher priority being executed if two or more schedule scripts overlap.

Further optionally, each of the plurality of schedule scripts may include a start time and an end time, between which the images are to be captured, and a designation of days of a week, on which the images are to be captured.

Still optionally, each of the plurality of schedule scripts may further include an interval, at every occurrence of which an image is to be captured.

According to a further aspect of the invention, there is provided an Internet camera that captures images of an object and transmitting image data to a predetermined site

on the Internet, which camera is provided with a memory that stores schedule data which includes a plurality of schedule scripts, a network interface connected to the Internet, a data transfer client that controls the network interface to transfer the image data files as stored to a predetermined site on the Internet, a schedule merging device that merges the plurality of schedule scripts into a merged schedule, and a controller that controls the data transfer client to transfer the stored image data files in accordance with the merged schedule to the predetermined site.

With the above configuration, since the image data transferring operations can be performed in accordance with a merged schedule which is generated by combining a plurality of schedule scripts, a complicated scheduling can be achieved easily.

Optionally, the schedule merging device may assign priorities to the plurality of schedule scripts, respectively, procedures defined by a schedule script having a higher priority being executed if two or more schedule scripts overlap.

Further optionally, each of the plurality of schedule scripts may include a start time and an end time, between which the stored image data files are to be transferred, and a designation of days of a week, on which the stored

image data files are to be transferred.

In this case, each of the plurality of schedule scripts may further include an interval, at every occurrence of which the stored image data files are to be transferred.

Brief Description of the Accompanying Drawings

Fig. 1 is a block diagram of an Internet camera according to an embodiment of the present invention;

Figs. 2A-2C show a structure of the schedule data;

Fig. 3 is a flowchart showing an operation in the embodiment of the present invention;

Fig. 4 is a flowchart showing the detailed operation of step S8 of Fig. 3;

Fig. 5 is an example of a schedule data of the embodiment of the present invention; and

Figs. 6 is another example of a schedule data of the embodiment of the present invention.

Detailed Description of the Embodiments

Fig. 1 is a block diagram of an Internet camera according to an embodiment of the present invention. The Internet camera 1 includes an image capturing device 2, a

storage device 3, a controller 6 including a CPU, a memory 7, an FTP client 5, and a network interface 4, such as a modem. It should be noted that the network interface may include a LAN card interface connectable to the Internet through a proxy server.

In the Internet camera 1, the controller 6 controls the image capturing device 2 that captures an image and obtains image data. The image data is stored in the storage device 3 as an image data file. The storage device 3 is capable of storing a plurality of image data files captured at different timings.

The FTP client 5, which is controlled by the controller 6, controls the network interface 4 to transfer the image data files stored in the storage device 3 to a predetermined site of an FTP server (not shown) on the Internet.

In the memory 7, schedule data is stored. Specifically, the schedule data is stored in a form of a data file, which may contain one or more schedule scripts. The controller 6 retrieves the schedule data, and controls the Internet camera 1 to capture images and transfer the image data files in accordance with the schedule script(s) included in the schedule data. It should be noted that, in the embodiment, an image is captured and then an image data file corresponding to the captured image is transmitted to

the predetermined site immediately, in accordance with the schedule data. However, the control may be modified such that only one of the image capturing operation or data transmitting operation is executed in accordance with the schedule data, or the image capturing operation and the image data transmitting operation are executed in accordance with different schedules, respectively.

Figs. 2A-2C schematically show an example of the structure of the schedule data. Fig. 2A shows a data structure of the schedule data. The schedule data is stored from a predetermined address (START-ADDRESS) of the memory 7. In the example shown in Fig. 2A, the schedule data includes a plurality of schedule scripts, i.e., a first through N-th schedule scripts. Fig. 2B shows an example of a structure of a schedule script. As shown in Fig. 2B, each schedule script includes a DAY-FLAG parameter, a START-TIME parameter, an END-TIME parameter, an INTERVAL parameter, and a NEXT-POINTER parameter.

Fig. 2C shows a structure of the DAY-FLAG parameter. As shown in Fig. 2C, the DAY-FLAG parameter is a 7-bit parameter. Each of the bits corresponds to each day of a week. That is, from the LSB (Least Significant Bit) of the DAY-FLAG parameter, the bits of the DAY-FLAG parameter represent the TRUE/FALSE states for Monday, Tuesday, Wednesday, Thursday, Friday, Saturday and Sunday,

respectively.

If a bit of the DAY-FLAG parameter corresponding to a certain day is "TRUE", at least one capturing operation is executed on the day, between a start time defined by the START-TIME parameter and an end time defined by the END-TIME parameter.

Specifically, the first capturing operation on the day, whose status is "TRUE", is executed at the start time that is defined by the START-TIME.

The INTERVAL parameter indicates an interval. The image capturing and transferring operations are to be executed at the intervals indicated by the INTERVAL parameter, between the start time and the end time. For example, if the START-TIME parameter indicates "11:00", the END-TIME parameter indicates "13:00" and the INTERVAL parameter indicates "30 minutes", images are captured and transferred at 11:00, 11:30, 12:00, 12:30 and 13:00.

The NEXT-POINTER parameter represents a top address in the memory 7 at which the next schedule script is stored if it is included in the schedule data. If no further schedule script is included in the schedule data, the value of the NEXT-POINTER parameter is set to "Null".

Fig. 3 shows an image capturing/transferring procedure according to the embodiment of the present invention. The procedure uses a READ-POINTER variable, a CURRENT-TIME

variable, a CURRENT-DAY variable, a LAST-TRANSFERRED variable, and a SCHEDULE-BUFFER variable. In the following description, {} represents a data value of the variables. For example, {CURRENT-TIME} represents the time set to the CURRENT-TIME variable.

In S0, the LAST-TRANSFERRED variable is set to "Null", and in S1, the READ-POINTER is set to {START-ADDRESS}, i.e., the value of the START-ADDRESS.

Then, in S2, a schedule script, which starts from an address indicated by the Read-Pointer, is copied to the SCHEDULE-BUFFER. Then in step S3, it is judged whether the schedule script copied in the SCHEDULE-BUFFER is corrupted. If the schedule script in the SCHEDULE-BUFFER is not corrupted (S3: NO), then, in S4, the CURRENT-TIME and CURRENT-DAY variables are set to the current time and current day, respectively. In the embodiment, the current time and current day may be obtained from a time-server on the Internet using a Network Time Protocol. Alternatively, the current time and the current day are obtained from a real-time clock (not shown) of the controller 6. In such a case, the real-time clock may be adjusted based on the time and day obtained from the time-server on the Internet.

In S5, the status of a bit of the DAY-FLAG corresponding to the CURRENT-DAY is set to "TRUE" or "FALSE". If the bit is "TRUE" (S5: TRUE), then, in S6, it

is judged whether the {CURRENT-TIME} is after the {START-TIME} of the SCHEDULE-BUFFER. If the {CURRENT-TIME} is after the {START-TIME} (S6: YES), then, in S7, it is judged whether the {CURRENT-TIME} is on or before the {END-TIME} of the Schedule-Buffer.

If the {CURRENT-TIME} does not exceed the {END-TIME} (S7: NO), then, in S8, a procedure for capturing images, storing image data files converted from the captured image, and transferring the image data files is executed at intervals defined by the INTERVAL parameter. Then, control returns to S1, and the image capturing/transferring operation according to another schedule script will be executed.

If the {SCHEDULE-BUFFER} is determined to be corrupted (S3: YES), then, in S11, an error message is output. The error messages may be transferred, as an error message file, to the predetermined site, through the FTP client 5, in order to inform a client remote from the Internet camera 1 of the fact the a schedule script is corrupted. Alternatively or optionally, the error message may be transferred to a user as an e-mail message through a not-shown SMTP client. Then, in order to copy the schedule script again from the memory 7 to the SCHEDULE-BUFFER, control returns to S1 and the setting of the pointer at S1 and the data-loading operation at S2 are executed again.

If the bit corresponding to the CURRENT-DAY is set to "FALSE" in S5, then, in S21, it is judged whether the {NEXT-POINTER} of the SCHEDULE-BUFFER is "NULL". If the {NEXT-POINTER} is "Null" (S21: YES), then, in S22, the LAST-TRANSFERRED variable is cleared, and control returns to S1. If the {NEXT-POINTER} is not "Null" (S21: NO), then, in S31, the READ-POINTER is set to the {NEXT-POINTER}, and control returns to S2, at which another image capturing/transferring operations according to the next schedule script will be executed.

If the {CURRENT-TIME} does not exceed the {START-TIME} (S6: NO), then, in S21, it is judged whether the {NEXT-POINTER} is "Null".

If the {CURRENT-TIME} exceeds the {END-TIME} (S7: YES), then, in S21, it is judged whether the {NEXT-POINTER} of the {SCHEDULE-BUFFER} is "Null".

When the {CURRENT-TIME} is outside the programmed time in the current schedule, controls proceeds from S6 or S7 to S21 to judge whether the {NEXT-POINTER} is "Null". If the {NEXT-POINTER} is not "Null", control returns to S2 vial S31 to operate in accordance with the next schedule. Therefore, when the schedules do not overlap in programmed time for the same day, each schedule is accommodated. This permits at least day-spanning merged schedules. Even if more than two schedule scripts are provided, unless the

schedules do not overlap in programmed time for the same day, all the schedules can be accommodated.

Further, when the {CURRENT-TIME} is within the programmed time, control proceeds from S6 to S7, and then S8, and S31 is not executed. In other words, control does not check the next schedule until the {CURRENT-TIME} is outside the programmed time in the current schedule. Therefore, if the schedules overlap in programmed time for the same day, the current schedule overrides or has priority over the next schedule. Accordingly, by appropriately assigning the priorities to the schedule scripts, at least variable-interval schedules can be performed.

Fig. 4 is a flowchart showing the detailed procedure of step S8 of Fig. 3. In S101, the CURRENT-TIME variable is set to the current time. Then in S102, it is judged whether the {LAST-TRANSFERRED} variable is "Null". If the {LAST-TRANSFERRED} is "Null" (S102: YES), then, in S103, the {LAST-TRANSFERRED} variable is set to the {CURRENT-TIME}, i.e., the value (i.e., time) of the CURRENT-TIME variable. Then in S104, the controller 6 controls the image capturing device 2, the storage device 3 and the FTP client 5 to capture an image, convert the captured image into image data, store the image data in the storage device 3 as an image data file, then transfer the image data file to the

predetermined site.

If the LAST-TRANSFERRED variable is not "Null" (S102: NO), then, in S111, it is judged whether the value "{LAST-TRANSFERRED} + {INTERVAL}" exceeds 24:00 of the day. If the time does not exceed 24:00 of the day (S111: YES), then, in S112, it is judged whether the time "{LAST-TRANSFERRED} + {INTERVAL}" exceeds the {CURRENT-TIME}. If the time "{LAST-TRANSFERRED} + {INTERVAL}" exceeds the {CURRENT-TIME} (S112: YES), then, in S103, the LAST-TRANSFERRED variable is set to the {CURRENT-TIME}, i.e., the value (i.e., time) of the CURRENT-TIME. Then, in S104, an image is captured, converted into image data, stored in the memory 7 as an image data file, and the image data file is transferred to the predetermined site. It should be noted that, since the LAST-TRANSFERRED variable is set to the {CURRENT-TIME}, steps S103 and S104 are executed once at every interval between the start time and end time.

In step S111, if the time represented by "{LAST-TRANSFERRED} + {INTERVAL}" exceeds 24:00 of the day (S111: YES), then, the procedure shown in Fig. 4 is terminated.

In S112, if the time represented by "{LAST-TRANSFERRED} + {INTERVAL}" exceeds the CURRENT-TIME (S112: YES), then, the procedure shown in Fig. 4 is terminated.

Exemplary Schedule Data

Fig. 5 is an example of the schedule data consisting of first and second schedule scripts. According to the first schedule script, images are captured and transferred at every 15-minute interval, from 0:00 to 8:00, on Tuesday, Wednesday, Thursday and Friday.

According to the second schedule script, the images are captured and transferred at 15-minute intervals, from 21:00 to 24:00, on Monday, Tuesday, Wednesday and Thursday.

In accordance with the procedure shown in Figs. 3 and 4, the two schedule scripts are merged, and therefore the images are captured and transferred, at 15-minute intervals, from 21:00 of Monday, Tuesday, Wednesday and Thursday to 8:00 of the next day, respectively. Thus, according to the embodiment, a schedule which extends in two days can easily be realized.

Fig. 6 is another example of the schedule data which consists of two schedule scripts. According to the first schedule script, the images are captured and transferred at 10-minute intervals, from 8:00 to 17:00, on Monday, Tuesday, Wednesday, Thursday and Friday.

According to the second schedule script, the images are captured and transferred everyday, at 30-minute intervals, from 0:00 to 24:00.

According to the embodiment, as understood from the flowchart shown in Fig. 3, priorities are assigned to the

schedule scripts: the first schedule script has a higher priority than the second schedule script. Therefore, during the time period, which is referred to by both of the first and second schedule scripts, only the first schedule script is valid, and the second schedule script is invalidated.

Therefore, in the second example, the operation according to the second schedule script will not be executed from 8:00 to 17:00 of Monday through Friday. That is, on Monday, Tuesday, Wednesday, Thursday and Friday, the images are captured and transferred at 30-minute intervals from 0:00 to 8:00, at 10-minute intervals from 8:00 to 17:00, and at 30-minute intervals from 17:00 to 24:00, and on Saturday and Sunday, images are captured and transferred at 30-minute intervals, from 0:00 to 24:00.

As above, according to the present invention, by combining a plurality of schedule scripts, each having a relatively simple data structure, a complicated scheduling can be achieved easily.

It should be noted that the present invention is not restricted to the embodiment and examples described above. For instance, the camera 1 may not use the FTP client 5 and a network interface 4 (e.g., a modem), so that the images are merely captured and stored in the storage device 3 as image data files, according to the schedule data. Since the storage device 3 is capable of storing a plurality of image

data files, the images captured according to the schedule data are observed by controlling the digital camera manually. Alternatively, the stored image data files may be transferred in accordance with a procedure which is different from the image capturing procedure.

The present disclosure relates to the subject matter contained in Japanese Patent Application No. 2000-041764, filed on February 18, 2000, which is expressly incorporated herein by reference in its entirety.